

LAYERING AS OPTIMIZATION DECOMPOSITION:

QUESTIONS And ANSWERS

Mung Chiang, Steven H. Low, A. Robert Calderbank, John C. Doyle

Abstract— Network protocols in layered architectures have historically been obtained on an ad-hoc basis, and much of the recent cross-layer designs are conducted through piecemeal approaches. Network protocols may instead be holistically analyzed and systematically designed as distributed solutions to some global optimization problems in the form of generalized Network Utility Maximization (NUM), providing insight on what they optimize and on the structures of network protocol stacks.

In the form of 10 Questions and Answers, this paper presents a short survey of the recent efforts towards a systematic understanding of “layering” as “optimization decomposition”. The overall communication network is modeled by a generalized NUM problem, each layer corresponds to a decomposed subproblem, and the interfaces among layers are quantified as functions of the optimization variables coordinating the subproblems. Furthermore, there are many alternative decompositions, each leading to a different layering architecture. Industry adoption of this unifying framework has also started. Here we summarize the current status of horizontal decomposition into distributed computation and vertical decomposition into functional modules such as congestion control, routing, scheduling, random access, power control, and coding. We also discuss under-explored future research directions in this area.

More importantly than proposing any particular cross-layer design, this framework is working towards a mathematical foundation of network *architectures* and the design process of modularization.

Keywords: Adaptive coding, Cross-layer design, Congestion control, Distributed algorithm, Lagrange duality, MAC, Network architecture, Network utility maximization, Optimization, Power control, Reverse engineering, Routing, TCP/IP, Scheduling, Stochastic control, Wireless ad hoc networks.

I. OVERVIEW

In order to practically realize the potential of network-centric technology, system-level analysis and design for

performance and robustness must become far more systematic than the state-of-the-art, and ideally should be based on a firm theoretical foundation and first principles. While the individual subjects of communications, control, and computation have rich theories of practical relevance, they are fragmented and incompatible. Network-level design must coherently integrate all of these issues and until recently has been largely ad hoc and heuristic. This is inadequate both for the evolution of existing network infrastructure and the successful deployment of future, advanced network-centric technologies. We argue that the viewpoint of “architecture first” is essential and outline the progress and promise of a new framework of “layering as optimization decomposition”.

The idea that network architectures may be mathematically understood is a bold one. Network *architecture* determines *functionality* allocation: “who does what” and “how to connect them”, rather than just *resource* allocation. Layered architectures form one of the most fundamental and influential structures of network design. It adopts a modularized and often distributed solution approach to network coordination and resource allocation. Each layer controls a subset of the decision variables, and observes a subset of constant parameters and the variables from other layers. Intuitively, layered architectures enable a scalable, evolvable, and implementable network design while introducing potential risks to manageability of the network. There is clearly more than one way to “divide and conquer” the network design problem. For example, from a data-plane performance point of view, some layering schemes may be more efficient or more fair than others. Examining the choices of modularized design for networks, we would like to tackle the question of “how to” and “how not to” layer. Most of the recent work focus on resource allocation functionalities and performance metrics only. The limitations of such focus will also be discussed here.

Each layer in the protocol stack hides the complexity of the layer below and provides a service to the layer above. While the general principle of layering is widely recognized as one of the key reasons for the enormous success of data networks, there is little quantitative understanding to guide a systematic, rather than an ad hoc, process

M. Chiang is with Electrical Engineering Department, Princeton University, chiangm@princeton.edu. S. H. Low is with Computer Science and Electrical Engineering Departments, Caltech, slow@caltech.edu. A. R. Calderbank is with Electrical Engineering and Mathematics Departments, Princeton University, calderbk@princeton.edu. J. C. Doyle is with Control and Dynamic Systems Department, Caltech, doyle@cds.caltech.edu.

of designing layered protocol stack for wired and wireless networks. One possible perspective to rigorously and holistically understand layering is to integrate the various protocol layers into a single coherent theory, by regarding them as carrying out an asynchronous distributed computation over the network to implicitly solve a global optimization problem. Different layers iterate on different subsets of the decision variables using local information to achieve individual optimality. Taken together, these local algorithms attempt to achieve a global objective. Such a design process of modularization can be quantitatively understood through the mathematical language of *decomposition theory* for constrained optimization [51].

Such a framework of “layering as optimization decomposition” exposes the interconnection between protocol layers and can be used to study rigorously the performance tradeoff in protocol layering, as different ways to modularize and distribute a centralized computation. Even though the design of a complex system will always be broken down into simpler modules, this theory will allow us to systematically carry out this layering process and explicitly trade off design objectives.

The key idea in “layering as optimization decomposition” is as follows. Different *vertical decompositions* of an optimization problem, in the form of a generalized Network Utility Maximization (NUM), are mapped to different *layering schemes* in a communication network, with each decomposed subproblem in a given decomposition scheme corresponds to a layer, and functions of primal or Lagrange dual *variables* (coordinating the subproblems) correspond to the *interfaces* among the layers. *Horizontal decompositions* can be further carried out within one functionality module into distributed computation and control over geographically disparate network elements. Since different decompositions lead to alternative layering architectures, we can also tackle the question “how and how not to layer” by investigating the pros and cons of decomposition techniques. Furthermore, by comparing the objective function values under various forms of optimal decompositions and suboptimal decompositions, we can seek “separation theorems” among layers: conditions under which layering incurs no loss of optimality. Robustness of these separation theorems can be further characterized by sensitivity analysis in optimization theory: how much will the differences in the objective value (between different layering schemes) fluctuate as constant parameters in the generalized NUM formulation are perturbed.

There have been many recent research activities along the above direction by research groups around the world. In Section II, we use an innovative format in this paper to explain the “common language” of “layering as op-

timization decomposition”. We pose 10 skeptical questions that critically challenge the meaning and utility of the framework, and present the associated answers as we understand them today. Some of the answers are far from being complete, reflecting the fact this research area is a very active one that still needs much future work. Some key messages are summarized in Section III.

This short paper only provides a high-level tutorial of the collective efforts by many in the research community in this area. A more comprehensive survey can be found in [10].

II. TEN QUESTIONS AND ANSWERS

A. *Isn't this just another cross-layer design? What kind of architectural issues does it resolve?*

Answer: There are two intellectually fresh cornerstones behind “layering as optimization decomposition”. The first is “network as an optimizer”. The idea of “protocol as a distributed solution” to some global optimization problem (in the form of the basic NUM) has been successfully tested in trials for Transmission Control Protocol (TCP) [26]. The key innovation from this line of work [29], [34], [35], [44], [45], [47], [55] is to view TCP/IP network as an optimization solver and each variant of congestion control protocol as a distributed algorithm solving a specified basic NUM with a different utility function. In the basic NUM, the objective is the sum of source utilities as functions of rates, the constraints are linear flow constraints, and optimization variables are source rates. Other recent results also show how to reverse engineer Border Gateway Protocols (BGP) as solving the Stable Path Problem [20], and contention-based Medium Access Control (MAC) protocols as game-theoretic selfish utility maximization [38], [59]. Starting from a given protocol originally designed based on engineering heuristics, reverse engineering discovers the underlying mathematical problems being solved by the protocols and demonstrates the application of derived insights through forward engineering improvements of the protocols.

The second key concept is “layering as decomposition”. As will be discussed in the answers to the next two questions, generalized NUM problems can be formulated to represent a network design problem. These generalized NUM problems put the end user utilities at the “driver’s seat” for network design. For example, benefits of innovations in physical layers through better modulation and coding schemes are now characterized by the enhancement to applications rather than just the drop in bit error rates, which the users do not directly observe. An optimal solution to a generalized NUM formulation automatically establishes the benchmark for all layering schemes. The

problem itself does not have any pre-determined layering architecture. Indeed, layering is a human engineering effort.

The overarching question then becomes how to attain an optimal solution to a generalized NUM in a modularized and distributed way. Vertical decompositions across modules and horizontal decompositions across disparate network elements can be conducted systematically through the theory of decomposition for nonlinear optimization. Implicit message passing (where the messages have physical meanings and need to be measured anyway) or explicit message passing quantifies the amount of information sharing and decision coupling required for a particular decomposition.

There are many different ways to decompose a given problem, each of which corresponds to a different layering architecture. Even a different representation of the same NUM problem can lead to different decomposability structures even though the optimal solution remains the same. These decompositions, *i.e.*, layering schemes, have different characteristics in efficiency, robustness, asymmetry of information and control, and tradeoff between computation and communication. Some are “better” than others depending on the criteria set by the network users and managers. A systematic exploration in the space of alternative decompositions is possible, where each particular decomposition represents a holistically designed protocol stack.

Given the layers, crossing layers is tempting. For example, layers can be crossed for wired or wireless networks in many different ways. As evidenced by the large and ever growing number of papers on cross layer design over the last few years, we expect that there will be no shortage of cross layer ideas based on piecemeal approaches. The growth of the “knowledge tree” on cross layer design has been exponential. However, any piecemeal design jointly over multiple layers does not bring more structured thinking process than the ad hoc design of just one layer. What seems to be lacking is a level ground for fair comparison among the variety of cross layer designs, a unified view on how to and how not to layer, and fundamental limits on the impacts of layer-crossing on network performance and robustness metrics.

“Layering as optimization decomposition” provides a candidate for such a unified framework. It attempts at shrinking the “knowledge tree” on cross layer design rather than growing it. It is important to note that “layering as optimization decomposition” is *not* the same as the generic phrase of “cross-layer optimization”. What is unique about this framework is that it views the network as the optimizer itself, puts the end user application needs as

the optimization objective, provides the globally optimal performance benchmark, and leads to a systematic design of decomposed solutions to attain the benchmark. Not all architectural principles may be readily quantified, but “layering as optimization decomposition” provides one of the few promising directions towards a mathematical theory of network architectures.

B. How to pick utility objective functions? How to guarantee QoS?

Answers: First of all, in reverse engineering, utility functions are implicitly determined by the given protocols already, and are to be discovered rather than designed. In forward engineering, utility functions can be picked based on any of the following four considerations:

- First, elasticity of application traffic can be modeled through utility functions.
- Second, utility can be defined by human psychological and behavioral models such as Mean Opinion Score in voice applications.
- Third, utility functions provide a metric to define optimality of resource allocation efficiency.
- Fourth, different shapes of utility functions lead to optimal resource allocations that satisfy some definition of fairness (*e.g.*, α -fair utilities parameterized by $\alpha > 0$: $U(x) = (1 - \alpha)^{-1}x^{1-\alpha}$ [47] leads to α -fair resource allocation).

In general, depending on who is interested in the outcome of network design, there are two types of objective functions: sum of utility functions by end users, which can be functions of rate, reliability, delay, jitter, or power level, and a network-wide cost function by network operators, which can be functions of congestion level, energy efficiency, network lifetime, or collective estimation error. Utility functions can be coupled across the users, and may not have an additive structure (*e.g.*, network lifetime).

Maximizing a weighted sum of all utility functions is only one of the possible formulations. An alternative is multi-objective optimization to characterize the Pareto-optimal tradeoff between the user objective and operator objective. Another set of formulations is game-theoretic between users and operators, or among users or operators themselves.

While utility models lead to objective functions, the constraint set of a NUM formulation incorporates the following two types of constraints. First is the collection of physical, technological, and economic restrictions in the communication infrastructure. Second is the set of per-user, hard, inelastic QoS constraints that cannot be violated at the equilibrium. This is in contrast to the utility

objective functions, which may represent elastic QoS demands of the users.

C. What kind of NUM formulations are there?

Answers: The basic NUM problem is the following formulation [29], known as monotropic programming and studied since 1960s. TCP variants have recently been reverse engineered to show that they are implicitly solving this problem, where source rate vector \mathbf{x} is the only set of optimization variables, and routing matrix \mathbf{R} and link capacity vector \mathbf{c} are both constants:

$$\begin{aligned} & \text{maximize} && \sum_s U_s(x_s) \\ & \text{subject to} && \mathbf{R}\mathbf{x} \preceq \mathbf{c}. \end{aligned} \quad (1)$$

Utility functions U_s are often assumed to be smooth, increasing, concave, and dependent on local rate only, although recent investigations have removed some of these assumptions for applications where they are invalid.

Many of the papers on “layering as optimization decomposition” are special cases of the following generic problem [8], one of the possible formulations of a generalized NUM for the entire protocol stack:

$$\begin{aligned} & \text{maximize} && \sum_s U_s(x_s, P_{e,s}) + \sum_j V_j(w_j) \\ & \text{subject to} && \mathbf{R}\mathbf{x} \leq \mathbf{c}(\mathbf{w}, \mathbf{P}_e), \\ & && \mathbf{x} \in \mathcal{C}_1(\mathbf{P}_e), \mathbf{x} \in \mathcal{C}_2(\mathbf{F}) \text{ or } \in \Pi, \\ & && \mathbf{R} \in \mathcal{R}, \mathbf{F} \in \mathcal{F}, \mathbf{w} \in \mathcal{W}. \end{aligned} \quad (2)$$

Here, x_s denotes the rate for source s and w_j denotes the physical layer resource at network element j . The utility functions U_s and V_j may be any nonlinear, monotonic functions. \mathbf{R} is the routing matrix, and \mathbf{c} are the logical link capacities as functions of both physical layer resources \mathbf{w} and the desired decoding error probabilities \mathbf{P}_e . The issue of signal interference and power control can be captured in this functional dependency. The rates must also be constrained by the interplay between channel decoding reliability and other error control mechanisms like ARQ. This constraint set is denoted as $\mathcal{C}_1(\mathbf{P}_e)$. The issue of rate-reliability tradeoff and coding is captured in this constraint. The rates are further constrained by the medium access success probability, represented by the constraint set $\mathcal{C}_2(\mathbf{F})$ where \mathbf{F} is the contention matrix, or the schedulability constraint set Π . The issue of medium access control (either random access or scheduling) is captured in this constraint. The sets of possible physical layer resource allocation schemes, of possible scheduling or contention based medium access schemes, and of single-path or multi-path routing schemes are represented by \mathcal{W} , \mathcal{F} , \mathcal{R} , respectively. The optimization variables are \mathbf{x} , \mathbf{w} , \mathbf{P}_e , \mathbf{R} , \mathbf{F} . Holding some of the variables

as constants and specifying some of these functional dependencies and constraint sets will then lead to a special class of this generalized NUM formulation. As discussed in the answer to the last question, utility functions and constraint sets can be even richer.

A deterministic fluid model is used in the above formulations. Stochastic network utility maximization is an active research area, as discussed later in this section. Whether it is the basic, general, or stochastic NUM, there are three separate steps in the process: first formulate a specific NUM problem, then devise a modularized and distributed solution following a particular decomposition, and finally explore the space of alternative decompositions that provide a choice of layered protocol stack and coupling across the layers.

There are many remaining challenges in formulating NUM. For example, it is still difficult to fully incorporate BGP for inter-AS routing in the generalized NUM framework. Similarly, an optimization-based, unifying view on wireless ad hoc network routing is lacking. Much further work remains to be done to model utility functions in specific applications, especially inelastic, real-time applications such as VoIP and streaming media [25]. In a more refined physical/link layer model, the option of forwarding rather than re-encoding at intermediate nodes must be considered, as well as retransmission through ARQ.

D. Isn't it just all about dual decomposition?

Answers: The basic idea of decomposition is to divide the original large problem into smaller subproblems, which are then coordinated by a master problem by means of some kind of signalling. Most of the existing decomposition techniques can be classified into *primal decomposition* and *dual decomposition* methods¹. The former is based on decomposing the original primal problem, whereas the latter is based on decomposing the Lagrange dual of the problem. Primal decomposition methods have the interpretation that the master problem directly gives each subproblem an amount of resources that it can use; the role of the master problem is then to properly allocate the existing resources. In dual decomposition methods, the master problem sets the price for the resources to each subproblem which has to decide the amount of resources to be used depending on the price; the role of the master problem is then to obtain the best pricing strategy. Primal decomposition and dual decomposition can in fact be inter-changed by introducing auxiliary variables [51].

Almost all the papers in the vast, recent literature on NUM use a standard dual-based distributed algorithm.

¹This is not to be confused with primal-dual interior-point algorithm, or primal driven network control, or primal penalty function approach.

Contrary to the apparent impression that such a decomposition is the only possibility, there are in fact many alternatives to solve a given network utility problem in different but all distributed manners [52], including multi-level and partial decompositions. Each of the alternatives provides a possibly different network architecture with different engineering implications.

Alternative horizontal decomposition (i.e., distributed control across geographically disparate network elements) has been studied in [52]. Recent results on alternative vertical decomposition (i.e., modularized control over multiple functional modules or layers) scatter in an increasingly large research literature. For example, on the topic of joint congestion control, routing, and scheduling, different decompositions have been obtained in [1], [5], [16], [41], [57], on the topic of joint congestion control and random access, different decompositions have been obtained in [62], [39], and on the topic of rate control for network coding based multicast, different decompositions have been obtained in [43], [64], [65], [2], [67], [7]. A systematic treatise on this variety of vertical decompositions is an interesting research direction that will contribute to a rigorous understanding of the architectural choices of allocating functionalities to control modules.

Coupling for generalized NUM can happen not only in constraints, but also in the objective function, where the utility of source s , $U_s(x_s, \{x_i\}_{i \in I(s)})$, depends on both its local rate x_s and the rates of a set of other sources with indices in set $I(s)$. If U_s is an increasing function of $\{x_i\}_{i \in I(s)}$, this coupling models cooperation in a clustered system, otherwise it models competition such as power control in wireless network or spectrum management in DSL. Such coupling in the objective function can be decoupled [58] by first introducing auxiliary optimization variables and consistency equality constraints, thus shifting coupling in objective to coupling in constraints, then introducing “consistency prices” to decouple the consistency constraints. These consistency prices are iteratively updated through local message passing.

E. How do you know which decomposition to pick?

Answers: Even a different representation of the same primal problem may change the duality and decomposability structures even though it does not change the optimal solution. It remains an open issue how to systematically explore the space of alternative vertical and horizontal decompositions, and thus the space of alternative network architectures, for a given set of requirements on, e.g., rate of convergence, symmetry of computational load distribution, and amount of explicit message passing.

An intellectually bold direction for future research is to explore if both the *enumeration* and *comparison* of alternative decompositions, horizontally and vertically, can be carried out systematically or even be automated.

To enumerate the set of possible decompositions, one has to take into account that transformations of the problem (e.g., change of variable) may lead to new decomposability structure, or turn a seemingly non-decomposable problem into a decomposable one [21]. This would open the door to even more alternative decompositions, each of which has a different engineering implication to distributed and modularized network architecture.

To compare alternative decompositions and the associated variety of distributed algorithms, the following metrics all need to be considered: speed of convergence, the amount and symmetry of message passing for global communication, the distribution of local computational load, robustness to errors, failures, or network dynamics, the possibility of efficient relaxations and simple heuristics, and the ability to remain evolvable as the application needs change over time. Some of these metrics do not have any quantitative units of measurement, such as evolvability. Some do not have a universally agreed upon definition, such as the measure of *how* distributed an algorithm is. Some are difficult to analyze accurately, such as the rate of convergence. Application contexts lead to a prioritization of these possibly conflicting metrics, based on which, the “best” decomposition can be chosen from the range of alternatives.

Summarizing, there are three stages of conceptual understanding of an optimization/decomposition view of network architectures:

- First, layered and distributed network architectures can be rigorously understood as decompositions of an underlying optimization problem.
- Second, there are in fact many alternatives of decompositions and therefore alternatives of network architectures. Furthermore, we can systematically explore and compare such alternatives.
- Third, there may be a methodology to exhaustively enumerate all alternatives, to quantify various comparison metrics, and even to determine a priori which alternative is the best according to any given combination of comparison metrics.

Many issues in the third stage of the above list remain open for future research.

F. What about those nonconvex optimization formulations?

Answers: Nonconvex optimization formulations of generalized NUM may appear for four types of reasons.

First, non-concave utilities, such as sigmoidal utility. Second, non-convex constraint set, such as lower bounds on SIR as a function of transmit power vector, in the low-SIR regime of interference-limited networks. Third, integer constraints, such as those in single-path routing protocols. Fourth, convex constraint sets that would require a description length that grows exponentially with the number of variables, such as certain schedulability constraints in multihop interference models. In general, nonconvex optimization is difficult in theory and in practice.

For example, nonconvex optimization often has non-zero duality gaps. A non-zero duality gap means that the standard dual-based distributed subgradient algorithm, and in general dual decomposition approaches, may lead to suboptimal and even infeasible primal solutions and instability in cross layer interactions. This very difficult problem can be tackled through a combination of well-established and more recent optimization techniques (e.g., sum-of-squares programming [53] and geometric-sigomial programming [9]). For example, there have been three recent approaches to solve nonconcave utility maximization over linear constraints:

- [36] proposes a distributed, suboptimal heuristic (for sigmoidal utilities) called “self-regulating” heuristics, which is shown to avoid link congestion caused by sigmoidal utilities.
- [22] determines optimality conditions for the dual-based distributed algorithm to converge globally (for all nonlinear utilities). The engineering implication is that appropriate overprovisioning of link capacities will ensure global convergence of the dual-based distributed algorithm even when user utility functions are nonconcave.
- [19] develops an efficient but centralized method to compute the global optimum (for a wide class of utilities that can be transformed into polynomial utilities), using the sum-of-squares method.

There are at least three very different approaches to tackle the difficult issue of nonconvexity:

Go around nonconvexity: discover a change of variable that turns the seemingly nonconvex problem into a convex one, determine conditions under which the problem is convex or the KKT point is unique, or make approximations to make the problem convex.

Go through nonconvexity: use successive convex relaxations (e.g., Sum-of-squares, Signomial programming), utilize special structures in the problem (e.g., difference of convex, generalized quasiconcavity), or leverage smarter branch and bound methods.

Go above nonconvexity: observe that optimization problem formulations are induced by some underlying as-

sumptions on what the architectures and protocols should look like. By changing these assumptions, a different, much easier-to-solve or easier-to-approximate NUM formulations may result. This is referred to as *design for optimizability* [24], which concerns with redrawing architectures to make the resulting generalized NUM easier to solve, rather than *optimization* that tries to solve a given, possibly difficult NUM problem.

G. Isn't fluid model with infinite backlog too restrictive?

Answers: When sessions (i.e., flows, connections, end-users) arrive and depart, packets come in bursts, channels vary over time, and topology is subject to change, new formulations of stochastic NUM become necessary, presenting new challenges on stability and performance characterization. Most of the known results concern stochastic stability and validity of the deterministic fluid model, with little characterization on the distribution of utility or user-perceived delay induced by the distributions of stochastic models at various levels.

Session level. For Poisson arrivals of sessions with exponentially distributed file size, [3], [14], [46] showed that, for certain classes of utility functions under the time-scale separation assumption², the stability region of the basic NUM is the largest possible, which is the capacity region formed by the fixed link capacities in the deterministic NUM formulation. Then [40], [56] extended this stochastic stability result to the case without the time-scale separation assumption. Extensions have recently been carried out to other models [57], [69], with fluid limits and diffusion approximations proposed as well [31], [32]. Recent results have also established session-level stochastic stability for any filesize distribution with finite moments (e.g., see [11] and references therein).

Packet level. There have been three sets of results that appeared over the last two years: translating on-off HTTP session utility into transport layer TCP utility (mapping from microscopic to macroscopic model) [4], showing many-flow asymptotical validation of fluid model (justifying the transition from microscopic to macroscopic model) [15], [54], and demonstrating convergence behavior for stochastic noisy feedback [72].

Channel level. Channel variations offer both the challenge to prove stability/optimality for existing algorithms and the ability to do opportunistic transmission and scheduling. For example, in [5], *stability* and *optimality* are established for dual algorithms under channel-level stochastic for any convex optimization where the constraint set has the following structure: a subset of the

²Here time-scale separation means that the resource allocation algorithm converges before the number of sessions changes.

variables lie in a polytope and other variables lie in a convex set that varies according to an irreducible, finite-state Markov chain. “Layering as optimization decomposition” type of algorithms that only require instantaneous knowledge of the current channel state (*e.g.*, queue-lengths) remain stable and optimal (in the expected sense).

Topology level. Very little has been explored on this topic, which is important for battery based or highly mobile wireless ad hoc networks.

H. Is anyone actually going to use this framework?

Answers: The application of “layering as optimization decomposition” has been illustrated through many case studies carried out by various research groups in the last couple of years, generating considerable general insights in addition to the specific cross-layer designs.

In the terminology of the standard seven layer reference model, it is well-known that physical layer algorithms try to solve the data transmission problem formulated by Shannon: maximizing data rate subject to vanishing error probability constraints. Recent progress have put protocols in layers 2-4 of the standard reference model on a mathematical foundation as well:

- The congestion control functionality of TCP has been reverse engineered to be implicitly solving the basic NUM problem (1). While *heterogeneous* congestion control protocols do not solve an underlying NUM problem, its equilibrium and dynamic properties can still be analyzed through a vector field representation and Poincare-Hopf index theorem [60], [61], which show that sufficiently small “degree of heterogeneity” implies global uniqueness and local stability of network equilibrium.
- IGP of IP routing are known to be variants of shortest path routing solvers, and the policy-based routing protocol in BGP has recently been modeled as the solution to the Stable Path Problem [20].
- Scheduling based MAC protocols are known to be solving variants of maximum weight matching, and random access (contention based MAC) protocols have recently been reverse engineered as a non-cooperative selfish utility maximization game [38], [59].

Following is a *non-exhaustive* list of some of the recent publications using “layering as optimization decomposition”, with references given in the bibliography³. In all these cases, a NUM problem that is more complicated than the basic NUM represents a more general networking

problem encompassing more than congestion control, and some functions of the Lagrange dual variables act as the “layering variables”.

- Jointly optimal congestion control and adaptive coding or power control
- Jointly optimal congestion and contention control
- Jointly optimal congestion control and scheduling
- Jointly optimal routing and scheduling
- Jointly optimal routing and power control
- Jointly optimal congestion control, routing, and scheduling
- Jointly optimal routing, scheduling, and power control
- Jointly optimal routing, resource allocation, and source coding
- TCP/IP interactions and jointly optimal congestion control and routing
- Network lifetime maximization

Industry adoption of “layering as optimization decomposition” has already started. For example, insights from reverse-engineering TCP has lead to an improved version of TCP implemented over the last several years: FAST (Fast AQM Scalable TCP) [17], [26]. Putting end-user application utilities as the objective function has lead to a new way to leverage innovations in the physical and link layers beyond the standard metrics such as bit error rate, *e.g.*, in “FAST Copper” Project (here FAST stands for Frequency, Amplitude, Space, Time) for an order-of-magnitude boost to rates in fiber/DSL broadband access systems [18].

I. Who cares about convergence at time infinity under a weird stepsize?

Answers: Understanding practical stepsize choices’ impacts in an asynchronous environment, characterizing transient behaviors of iterative algorithms, and tightly bounding the rate of convergence are all important and under-explored topics in this area.

For example, for certain applications, if the resource allocation (*e.g.*, window size, signal-to-interference-ratio) for a user drops below a threshold during the transient, the user may be disconnected. In such cases, the whole idea of equilibrium becomes meaningless. *Invariance* during transients, instead of convergence in the asymptote, becomes a more useful concept: how fast can the algorithm gets close enough to the optimum and stay in that region? Usually the overall system performance derived out of a modularized design determines “how close is close enough” for each module’s transients.

Similarly, in the objective function, utility as a function of delay, jitter, and even the entire time series of resource

³We apologize in advance for any references we may have missed and would appreciate any information about other citations.

allocation during the transients needs to be further investigated.

J. Why should network operators optimize performance in the first place?

Answers: Indeed, optimality is *not* the key point. Optimization is used here primarily as a modeling language and a starting point to develop and compare architectural choices, rather than defining a particular point of operation at global optimum. Suboptimal, but simple (low spatial-temporal complexity) algorithms can be used in various modules (e.g., the scheduling module), and, as long as the suboptimality gap is bounded and the network architecture is “good”, then the “damage” from the suboptimal design in one layer can be contained at the systems level [41]. Similarly, stochastic dynamics may also wash away the corner cases and be beneficial to the average system performance, if the network architecture is appropriately designed. In such cases, it is interesting to study the meaning of utility-suboptimality in terms of degradation to fairness.

Protocols and layered architectures are not just for maximizing the efficiency of performance metrics, such as throughput, latency, and distortion, but also robustness metrics, such as evolvability, scalability, availability, and manageability. Interactions among layers introduce the risks of losing robustness against unforeseen demands arising over time or significant growth over space. Despite the importance in practical network operations, these network X-ities remain as important yet fuzzy notions, and a quantified foundation for them is long overdue [12]. Intuitively, “design by decomposition” enhances scalability and evolvability, but may present risks to manageability such as diagnosability and optimizability. This is in part because layering means that each layer is limited in what it can do (optimization variables in a decomposed subproblem) and what it can observe (a subset of constant parameters and variables in other decomposed subproblems). Quantifying network X-ities, and trading-off network X-ities with performance metrics, in layered protocol stack design is a long-term, challenging direction.

Carrying the intellectual thread from “forward engineering” (solve a given problem) to “reverse engineering” (find the problem being solved by a given protocol) one step further to “design for optimizability”, it may be that the difficulty of solving a particular set of subproblems also illustrates that the given decomposition was conducted possibly in a wrong way and suggests that better alternatives exist.

Under the metrics of deployment cost and operations cost, the success of IP networks comes down to the scal-

ability and evolvability of TCP/IP and the way control is distributed and modularized. A long-term goal of the framework of “layering as optimization decomposition” is to provide a simple, relevant abstraction of what makes a network successful in this sense.

III. KEY MESSAGES

In summary, “layering as optimization decomposition” is a unifying framework for understanding and designing distributed control and cross-layer resource allocation in wired and wireless networks. It has been developed by various research groups over the last several years, and is now emerging to provide a mathematically rigorous and practically relevant approach to quantify the risks and opportunities of modifying existing layered network architecture. It shows that network protocols in layers 2, 3, and 4 can be reverse-engineered as implicitly solving some optimization-theoretic or game-theoretic problems. By distributively solving generalized NUM formulations through decomposed subproblems, we can systematically generate layered protocol stacks. There are many alternatives for both horizontal decomposition into disparate network elements and vertical decomposition into functional modules (*i.e.*, layers). While queuing delay or buffer occupancy is often used as the “layering price”, it may sometimes lead to unstable interactions. A variety of techniques to tackle coupling and nonconvexity issues have become available.

Some of the key messages obtained from many case studies are outlined below, and, as briefly discussed in this tutorial paper, even more open problems and new directions present themselves in this emerging area of research.

- Protocols in layers 2,3,4 can be *reverse engineered*. Reverse engineering in turn leads to better design in a rigorous manner.
- There is a *unifying approach* to cross-layer design, as summarized in Section I of this paper.
- Loose coupling through “layering price” can be optimal, and congestion price (or queuing delay, or buffer occupancy) is often the right “layering price” for stability and optimality, with important *exceptions* as well.
- There are many *alternatives* in decompositions, leading to different divisions of tasks across layers and even different time-scales of interactions.
- *Convexity* of the generalized NUM is the key to devising a *globally optimal* solution.
- *Decomposability* of the generalized NUM is the key to devising a *distributed* solution.

- *Architecture*, rather than optimality, is the most important theme.

More than just an ensemble of specific cross-layer designs for existing protocol stacks, “layering as optimization decomposition” is a *mentality* that views networks as optimizers, a *common language* that allows researchers to quantitatively compare alternative network architectures, and a suite of *methodologies* that facilitates a systematic design approach for modularized network architectures.

ACKNOWLEDGEMENT

We would like to gratefully acknowledge the collaborations and interactions on this topic with many colleagues, including Stephen Boyd, Ma’ayan Bresler, Lijun Chen, Dah-Ming Chiu, Neil Gershfeld, Andrea Goldsmith, Prashanth Hande, Jiayue He, Sanjay Hegde, Maryam Fazel, Bob Fry, Cheng Jin, Koushik Kar, Frank Kelly, P. R. Kumar, Jang-Won Lee, Ruby Lee, Lun Li, Ying Li, Xiaojun Lin, Jiaping Liu, Zhen Liu, Asuman Ozdaglar, Daniel Palomar, Pablo Parrilo, Jennifer Rexford, Devavrat Shah, Sanjay Shakkottai, Ness Shroff, R. Srikant, Chee Wei Tan, Ao Tang, Jiantao Wang, Xin Wang, David Wei, Bartek Wydrowski, Lin Xiao, Edmund Yeh, and Junshan Zhang.

The work summarized in this paper has been in part supported by NSF grants CCF-0440043, CCF-0448012, CNS-0417607, CNS-0427677, CNS-0430487, CNS-0519880, DARPA grant HR0011-06-1-0008, AFOSR FA9550-06-1-0297, and Cisco grant GH072605.

REFERENCES

- [1] M. Andrews, “Joint optimization of scheduling and congestion control in communication networks,” in *Proc. of CISS*, 2006.
- [2] S. Bhadra, S. Shakkottai, and P. Gupta, “Min-cost selfish multicast with network coding,” *Proc. IEEE NetCod Workshop*, April 2006.
- [3] T. Bonald and L. Massoulié, “Impact of fairness on Internet performance,” *Proc. ACM Sigmetrics*, pp. 82-91, 2001.
- [4] C. S. Chang and Z. Liu, “A bandwidth sharing theory for a large number of HTTP-like connections,” *IEEE/ACM Trans. on Networking*, vol. 12, no. 5, Oct. 2004.
- [5] L. Chen, S. H. Low, M. Chiang, and J. C. Doyle, “Optimal joint congestion control, routing, and scheduling design for wireless ad hoc networks,” *Proc. IEEE INFOCOM*, April 2006.
- [6] L. Chen, S. H. Low, and J. C. Doyle, “Joint congestion control and medium access control for wireless ad hoc networks,” *Proc. IEEE INFOCOM*, March 2005.
- [7] L. Chen, T. Ho, S. H. Low, M. Chiang, and J. C. Doyle, “Rate control for multicasting with network coding,” Preprint, 2006.
- [8] M. Chiang, “Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion and power control” *IEEE J. Sel. Area Comm.*, vol. 23, no. 1, pp.104-116, Jan. 2005.
- [9] M. Chiang, “Geometric programming for communication systems,” *Trends and Foundations in Communications and Information Theory*, vol. 2, no. 1, pp. 1-156, Aug. 2005.
- [10] M. Chiang, S. H. Low, R. A. Calderbank, and J. C. Doyle, “Layering as optimization decomposition”, To appear in *Proceedings of IEEE*, Jan. 2007. A shorter version appeared in *Proc. Conf. Inform. Sciences and Systems*, March 2006.
- [11] M. Chiang, D. Shah, and A. Tang, “Stochastic stability of network utility maximization: General filesize distribution”, *Proc. Allerton Conference*, Sept. 2006.
- [12] M. Chiang and M. Yang, “Towards X-ities from a topological point of view: Evolvability and scalability”, *Proc. Allerton Conf.* Oct. 2004.
- [13] R. L. Cruz and A. Santhanam: “Optimal routing, link Scheduling, and power control in multihop wireless networks,” *Proc. IEEE INFOCOM*, April 2003.
- [14] G. de Veciana, T. J. Lee, and T. Konstantopoulos, “Stability and performance analysis of network supporting elastic services”, *IEEE/ACM Trans. on Networking*, vol. 9, no. 1, pp. 2-14, Feb. 2001.
- [15] S. Deb, S. Shakkottai, and R. Srikant, “Asymptotic behavior of Internet congestion controllers in a many-flow regime”, *Math. Operations Research*, 2005.
- [16] A. Eryilmaz and R. Srikant, “Joint congestion control, routing and MAC for stability and fairness in wireless networks,” in *IEEE J. Sel. Area Comm.*, vol. 24, no. 8, Aug. 2006.
- [17] TCP FAST Project netlab.caltech.edu.
- [18] FAST Copper Project www.princeton.edu/fastcopper.
- [19] M. Fazel and M. Chiang, “Nonconcave network utility maximization through sum of squares method”, *Proc. IEEE CDC*, Dec. 2005.
- [20] T. G. Griffin, F. B. Shepherd, and G. Wilfong, “The stable path problem and interdomain routing,” *IEEE/ACM Trans. on Networking*, vol. 10, no. 2, pp. 232-243, April 2002.
- [21] P. Hande, S. Rangan, and M. Chiang, “Distributed power control for optimal QoS assignment in wireless data networks”, *Proc. IEEE INFOCOM*, April 2006.
- [22] P. Hande, S. Zhang, and M. Chiang, “Distributed rate allocation for inelastic flows”, Submitted to *IEEE/ACM Trans. on Networking*, preliminary version in *Proc. IEEE INFOCOM*, March 2005.
- [23] J. He, M. Chiang, and J. Rexford, “TCP/IP interaction based on congestion prices: Stability and optimality”, *Proc. IEEE ICC*, June 2006.
- [24] J. He, J. Rexford, and M. Chiang, “Design principles of manageable networks”, Preprint, 2006.
- [25] J. Huang, Z. Li, M. Chiang, and A. K. Katsaggelos, “Pricing-based rate control and joint packet scheduling for multi-user wireless uplink video streaming”, *Proc. IEEE Packet Video Workshop*, April, 2006.
- [26] C. Jin, D. X. Wei, and S. H. Low. TCP FAST: motivation, architecture, algorithms, performance. *Proc. of IEEE INFOCOM*, March 2004.
- [27] B. Johansson and M. Johansson, “Primal and dual approaches to distributed cross-layer optimization,” in *Proc. 16th IFAC World Congress*, Prague, Czech republic, 2005.
- [28] K. Kar, S. Sarkar, and L. Tassiulas, “Achieving proportional fairness using local information in Aloha networks,” *IEEE Trans. on Auto. Control*, vol. 49, no. 10, pp. 1858–1862, October 2004.
- [29] F. P. Kelly, A. Maulloo, and D. Tan, “Rate control for communication networks: Shadow prices, proportional fairness and stability,” *J. Op. Res. Soc.*, vol. 49, no. 3, pp.237–252, March 1998.
- [30] F. P. Kelly and T. Voice, “Stability of end-to-end algorithms for joint routing and rate control”, *Computer Communications Review*, vol. 35, no. 2, pp. 5-12, 2005.
- [31] F. P. Kelly and R. J. Williams, “Fluid model for a network op-

- erating under a fair bandwidth-sharing policy”, *Annals of Applied Probability*, vol. 14, no. pp. 1055-1083, 2004.
- [32] P. Key and L. Massoulie, “Fluid models of integrated traffic and multipath routing”, To appear in *Queueing Systems*, 2006.
- [33] M. Kodialam and T. Nandagopal, “Characterizing achievable rates in multi-hop wireless networks: The joint routing and scheduling problem”, *Proc. ACM Mobicom*, Sept. 2003.
- [34] S. Kunniyur and R. Srikant. End-to-end congestion control: utility functions, random losses and ECN marks. *IEEE/ACM Trans. on Networking*, 2003.
- [35] R. J. La and V. Anantharam, “Utility-based rate control in the Internet for elastic traffic,” *IEEE/ACM Trans. on Networking*, vol. 10, no. 2, pp. 272-286, April 2002.
- [36] J. W. Lee, R. R. Mazumdar, and N. Shroff, “Non-convex optimization and rate control for multi-class services in the Internet,” *Proc. IEEE INFOCOM*, Hong Kong, China, March 2004.
- [37] J. W. Lee, M. Chiang, and R. A. Calderbank, “Price-based distributed algorithm for optimal rate-reliability tradeoff in network utility maximization”, To appear *IEEE Journal of Selected Areas in Communications*, 2006.
- [38] J. W. Lee, M. Chiang, and R. A. Calderbank, “Utility-optimal medium access control: reverse and forward engineering,” *Proc. IEEE INFOCOM*, April 2006.
- [39] J. W. Lee, M. Chiang, and R. A. Calderbank, “Jointly optimal congestion and contention control in wireless ad hoc networks”, To appear in *IEEE Communication Letters*, 2006.
- [40] X. Lin and N. B. Shroff, “On the stability region of congestion control”, *Proc. Allerton Conference*, 2004.
- [41] X. Lin and N. Shroff, “Impact of imperfect scheduling in cross layer design,” *Proc. IEEE INFOCOM*, March 2005.
- [42] J. Liu, M. Chiang, and H. V. Poor, “Stability of distributed dual algorithm for convex optimization”, Preprint, 2006.
- [43] D. S. Lun, N. Ratnakar, M. Mardar, R. Koetter, D. R. Karger, T. Ho and E. Ahmed, “Minimum-cost multicast over coded packet networks”, *IEEE Trans. Inform. Theory*, 2005.
- [44] S. H. Low, “A duality model of TCP and queue management algorithms,” *IEEE/ACM Trans. on Networking*, vol. 11, no. 4, pp. 525-536, Aug. 2003.
- [45] S. H. Low and D. E. Lapsley, “Optimization flow control, I: basic algorithm and convergence,” *IEEE/ACM Trans. on Networking*, vol. 7, no. 6, pp. 861-874, December 1999.
- [46] L. Massoulie and J. W. Roberts, “Bandwidth sharing and admission control for elastic traffic”, *Telecommunication Systems*, vol. 15, pp. 185-201, March 2000.
- [47] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” *IEEE/ACM Trans. on Networking*, vol. 8, no. 5, pp. 556-567, Oct. 2000.
- [48] H. Nama, M. Chiang, and N. Mandayam, “Utility lifetime trade-off in self regulating wireless sensor networks: A cross-layer design approach”, *Proc. IEEE ICC*, June 2006.
- [49] T. Nandagopal, T. Kim, X. Gao, and V. Bharghavan, “Achieving MAC layer fairness in wireless packet networks,” *Proc. ACM Mobicom*, Boston, USA, Aug. 2000.
- [50] M. J. Neely, E. Modiano, and C. E. Rohrs, “Dynamic power allocation and routing time varying wireless networks”, *IEEE J. Sel. Area Comm.*, vol. 23, no. 1, pp. 89-103, Jan. 2005.
- [51] D. Palomar and M. Chiang, “A tutorial to decomposition methods for network utility maximization”, *IEEE J. Sel. Area Comm.*, vol. 24, no. 8, pp. 1439-1450, Aug. 2006.
- [52] D. Palomar and M. Chiang, “Alternative decompositions for distributed maximization of network utility: Framework and applications”, *Proc. IEEE INFOCOM*, April 2006.
- [53] P. A. Parrilo, “Semidefinite programming relaxations for semialgebraic problems,” *Mathematical Programming Series B*, vol. 96, no. 2, pp. 293-320, 2003.
- [54] S. Shakkottai and R. Srikant, “Mean FDE models for Internet congestion control under a many-flows regime”, *IEEE Trans. Inform. Theory*, vol. 50, no. 6, pp. 1050-1072, June 2004.
- [55] R. Srikant, *The Mathematics of Internet Congestion Control*, Birkhauser 2004.
- [56] R. Srikant, “On the positive recurrence of a Markov chain describing file arrivals and departures in a congestion-controlled network”, *IEEE Computer Communications Workshop*, 2004.
- [57] A. L. Stolyar, “Maximizing queueing network utility subject to stability: greedy primal-dual algorithm”, *Queueing Systems*, vol. 50, no. 4, pp. 401-457, 2005.
- [58] C. W. Tan, D. Palomar, and M. Chiang, “Decoupling coupled network utility maximization”, *Proc. IEEE ICASSP*, May 2006.
- [59] A. Tang, J. W. Lee, J. Huang, M. Chiang, and R. A. Calderbank, “Reverse engineering MAC”, *Proc. IEEE WiOpt*, April 2006.
- [60] A. Tang, J. Wang, S. H. Low, and M. Chiang, “Equilibrium of heterogeneous congestion control protocols”, *Proc. IEEE INFOCOM*, March 2005.
- [61] A. Tang, D. Wei, S. H. Low, and M. Chiang, “Heterogeneous congestion control: Efficiency, fairness, and control”, *Proc. IEEE ICNP*, Nov. 2006.
- [62] X. Wang and K. Kar, “Cross-layer rate control for end-to-end proportional fairness in wireless networks with random access,” *Proc. ACM Mobihoc*, May, 2005.
- [63] J. Wang, L. Li, S. H. Low, and J. C. Doyle. “Cross-layer optimization in TCP/IP networks,” *IEEE/ACM Trans. on Networking*, vol. 13, no. 3, pp. 582-595, August 2005.
- [64] Y. Wu and S.Y. Kung, “Distributed utility maximization for network coding based multicasting: A shortest path approach”, *IEEE Journal on Selected Areas in Communications*, Aug. 2006.
- [65] Y. Wu, M. Chiang, and S.Y. Kung, “Distributed utility maximization for network coding based multicasting: A critical cut approach”, *Proc. IEEE NetCod Workshop*, April 2006.
- [66] Y. Xi and E. Yeh, “Optimal distributed power control, routing, and congestion control in wireless networks”, Submitted to *IEEE Trans. Inform. Theory*, 2005.
- [67] Y. Xi and E. M. Yeh, “Distributed algorithms for minimum cost multicast with network coding”, *Proc. IEEE NetCod Workshop*, April 2006.
- [68] L. Xiao, M. Johansson, and S. Boyd, “Joint routing and resource allocation via dual decomposition,” *IEEE Trans. Comm.*, vol. 52, no. 7, pp. 1136-1144, July 2004.
- [69] H. Ye, “Stability of data networks under optimization-based bandwidth allocation”, *IEEE Trans. Auto. Control*, vol. 48, no. 7, pp. 1238-1242, July 2003.
- [70] W. Yu and J. Yuan, “Joint source coding, routing, and resource allocation for wireless sensor networks,” *Proc. IEEE ICC*, May 2005.
- [71] C. Yuen and P. Marbach, “Price-based rate control in random access networks,” *IEEE/ACM Trans. on Networking*, vol. 13, no. 5, pp. 1027-1040, December 2005.
- [72] J. Zhang, D. Zheng, and M. Chiang, “Impacts of stochastic noisy feedback in network utility maximization,” Preprint, 2006.